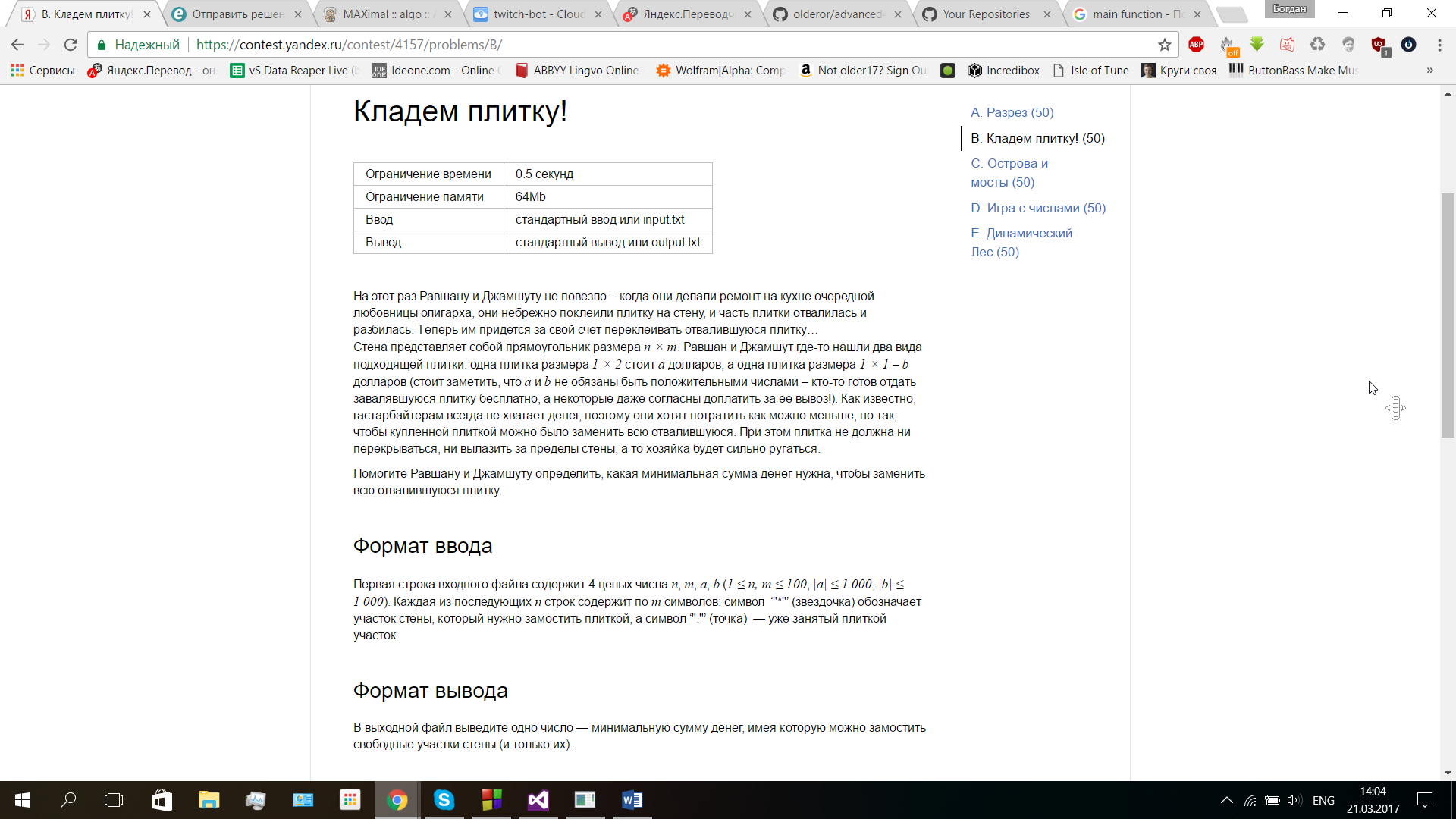
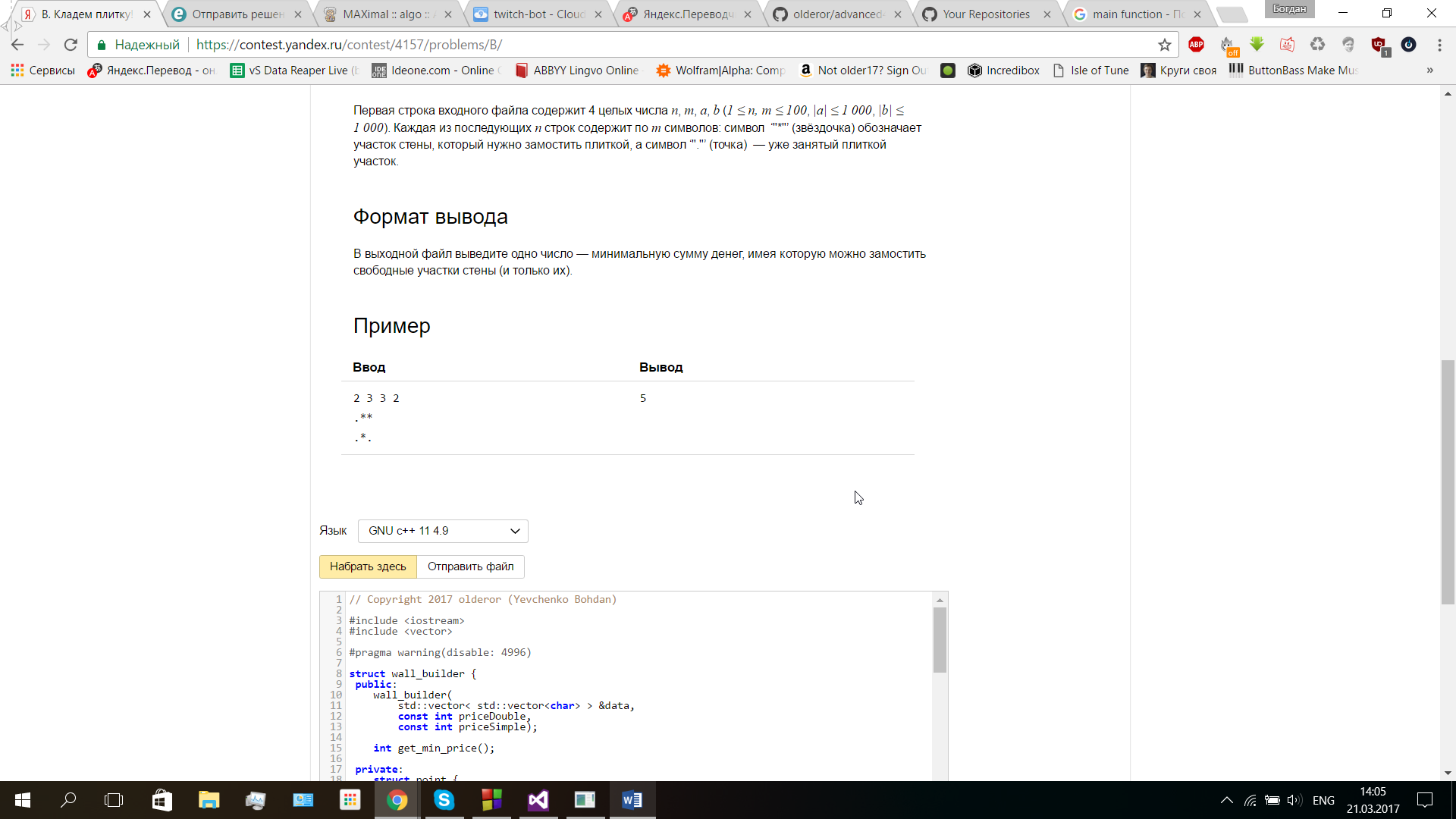
Задача «Кладем плитку!»

Решение Евченко Богдана ПИ-15-1

https://github.com/olderor/advanced4





Решение.

1. Если цена за 1х2 плитку выше, чем цена за 1х1 плитку, взятая два раза, то строим стену используя плитки 1х1. Ответ – цена за 1х1 плитку, умноженная на количество свободных мест.
2. Иначе построим стену с максимальным количеством плиток 1х2. Для этого строим двудольный граф. Пусть каждая свободная плитка будет вершиной в графе. Раскрасим поле на подобии шахматной доски и разобьем вершины по диагоналям. Таким образом каждая вершина из одной половины двудольного графа будет иметь соседей по ребру (вверх\вниз\влево\вправо) в другой половине двудольного графа (если соответствующая вершина свободная). Найдем максимальное паросочетание в таком графе – две вершины будут частями плитки 1х2. Количество ребер в таком паросочетании будет равно количеству плиток 1х2. Ответ – цена за 1х2, умноженная на количество ребер в паросочетании + цена за 1х1 для оставшихся клеток.

Доказательство

Если цена за 1х2 плитку выше, чем цена за 1х1 плитку, взятая два раза, то, очевидно, нужно брать каждую плитку отдельно, поскольку на плитку 1х2 мы потратим больше денег, чем на две 1х1.

Иначе нужно взять как можно больше плиток 1х2, поскольку каждая плитка 1х2 будет уменьшать общую стоимость. Для этого представим стену в виде двудольного графа, где каждая вершина будет связана с другой, если они являются соседями по ребру. Необходимо учесть, что плитки не могут пресекаться. Таким образом, данная задача сводится к задаче выбрать как можно больше рёбер, чтобы ни одно выбранное ребро не имело общей вершины ни с каким другим выбранным ребром – что является задачей поиска наибольшего паросочетания. Для этого подойдет алгоритм Куна, который работает за .

Итоги

Необходимо учесть, что в худшем случае каждый кусок стены является вершиной в двудольном графе. В каждой из половин двудольного графа будет вершин. Из каждой вершины может выходить максимум 4 ребра (поскольку вершина связана только с соседями сверху\снизу\слева\справа). Следовательно всего ребер . Алгоритм Куна работает за а значит он работает за . Также алгоритм Куна можно улучшить в несколько раз используя эвристики.

Оценка памяти. Для решения задачи потребовалось хранить состояние поля , хранить двудольный граф , для каждой вершины хранить значение – можно ли ее использовать . В целом это составляет

Оценка времени работы. Для решения задачи потребовалось: считать данные за , построить состояние поля за , построить двудольный граф за , найти наибольшее паросочетание за .

Итог:

* затраты по памяти.
* затраты по времени.